



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

ROZŠÍŘENÁ REALITA PRO ARTABLE S SEE-THROUGH BRÝLEMI

AUGMENT REALITY FOR ARTABLE AND SEE-THROUGH GLASSES

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

RÓBERT ADAM

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. MICHAL KAPINUS

BRNO 2017

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií

Akademický rok 2016/2017

Zadání bakalářské práce

Řešitel: **Adam Róbert**

Obor: Informační technologie

Téma: **Rozšířená realita pro ARTable s See-through brýlemi**
Augmented Reality for ARTable and See-Through Glasses

Kategorie: Uživatelská rozhraní

Pokyny:

1. Prostudujte koncept rozšířené reality a její využití při návrhu uživatelských rozhraní. Seznamte se s platfomou ROS a See-through brýlemi Vuzix.
2. Seznamte se s fakultní experimentální platformou ARTable, jejími možnostmi a aplikačním rozhraním.
3. Vyberte vhodné metody a nástroje a navrhnete uživatelské rozhraní, které umožní pomocí rozšířené reality zobrazení výběru objektů, nastavení parametrů a další úlohy potřebné na interaktivním stole.
4. Navržené rozhraní implementujte a integrujte do robotického systému ARTable.
5. Proveďte experimenty, demonstруйте a diskutujte vlastnosti vašeho řešení.
6. Vytvořte stručný plakát nebo video prezentující vaši bakalářskou práci, její cíle a výsledky.

Literatura:

- Dle pokynů vedoucího.

Pro udělení zápočtu za první semestr je požadováno:

- Splnění prvních třech bodů zadání.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Kapinus Michal, Ing.**, UPGM FIT VUT

Datum zadání: 1. listopadu 2016

Datum odevzdání: 17. května 2017

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
602 00 Brno, Božetěchova 2



doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstrakt

Táto bakalárska práca bola vytvorená počas môjho štúdia na fakulte informačných technológií Vysokého učení technického v Brne. Práca sa zaoberá rozšírenou realitou, vývojovým prostredím Unity a systémom ARTable. V teoretickej časti je obsiahnutá literatúra z oblasti histórie a použiteľnosti rozšírenej reality. Ďalej čo je to rozšírená realita a na akom princípe pracuje snímanie okolitého sveta. V praktickej časti sa práca zameriava na integráciu okuliarov Vuzix do Unity. Aplikácia dokáže na základe informácií poskytnutých od ARTable vykresliť do okuliarov rôzne údaje o objektoch nachádzajúcich sa na stole. Práca zahŕňa návrh riešenia a spôsob implementácie aplikácie. Na záver práca zahŕňa testovanie aplikácie a výsledok práce.

Abstract

This Bachelor Thesis was created during study at the Faculty of Information Technology at Brno University of Technology. The thesis is focused on augmented reality using Game Engine Unity and ARTable system. The theoretical part contains history and augmented reality usage. Then it is defined what augmented reality is and how camera tracking works in a real world. In a practical part the thesis is focused on integration of Vuzix glasses in to Unity. The application is able to draw information about objects placed on table differently according to information obtained from ARTable. The thesis describes design and implementation details about the created application. The final part of the thesis includes test results and conclusion.

Klíčové slová

Rozšírená realita, ARTable, ROS, Unity, Vuzix STAR 1200XL, ARToolKit

Keywords

Augmented reality, ARTable, ROS, Unity, Vuzix STAR 1200XL ARToolKit

Citácia

ADAM, Róbert. *ROZŠÍŘENÁ REALITA PRO ARTABLE S SEE-THROUGH BRÝLEMI*. Brno, 2017. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Michal Kapinus

ROZŠÍŘENÁ REALITA PRO ARTABLE S SEE-THROUGH BRÝLEMI

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením Ing. Michala Kapinuse. Dalšie informácie mi poskytli internetové a knižné zdroje. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....

Róbert Adam

17. mája 2017

Podakovanie

Moje úprimné podakovanie patrí môjmu konzultantovi Ing. Michalovi Kapinusovi z Ústavu počítačovej grafiky a multimédií, bez ktorého by táto práca nevznikla. Ďakujem mu za ochotu, odborný prístup, usmerňovanie a cenné rady pri konzultáciách.

Podakovanie patrí aj Bc. Ivici Mlynárovej za korektné upravenie práce do spisovného slovenského jazyka.

Obsah

1	Úvod	2
2	Teoretická časť	3
2.1	Rozšírená realita a jej história	3
2.2	Využitie	4
2.3	Zariadenia	8
2.4	Používateľské rozhranie	10
2.5	ARTable	11
2.6	Camera tracking	11
3	Návrh riešenia	13
3.1	Popis systémových častí	15
3.2	Výsledná scéna	17
3.3	Súradnicový systém	17
4	Implementácia	18
4.1	Implementačné nástroje	18
4.2	Testovanie	21
5	Záver	22
	Literatúra	23
A	Dodatočné súbory a predmety	24

Kapitola 1

Úvod

Rozšírená realita (ang. Augmented Reality, skr. AR) dokáže obohacovať reálny svet o grafické/virtuálne prvky. Pomocou aplikácie je teda možné sklbiť reálny a virtuálny svet. Na zobrazenie tých virtuálnych prvkov je potrebné zariadenie, ktoré podporuje rozšírenú realitu. V prípade vykresľovania grafických častí je potrebné, aby sa vykresľovali presne a preto musí byť aplikácia dobre navrhnutá. V tejto téze sa budem venovať práve rozšírenej realite a vývoju aplikácie podporujúcu AR.

Hlavným cieľom projektu je vytvoriť aplikáciu, ktorá bude zobrazovať základné informácie o objektoch. Ako som už spomínal je potrebné zariadenie, ktoré podporuje AR. V tejto práci budú týmto zariadením okuliare Vuzix STAR 1200 XL. Aplikácia bude vytvorená v prostredí Unity v operačnom systéme Windows 7. Výslednú aplikáciu je možné za pomoci Unity konvertovať aj do iných operačných systémov ako Linux a Mac OS.

Pri dosiahnutí cieľa som sa stretol s viacerými problémami. Niektoré z nich boli napríklad správne pripojenie spomínaných okuliarov k Unity. Zabezpečiť tak ich funkčnosť a správne zobrazovanie. Hlavnou úlohou okrem potrebných ovládačov bola kalibrácia zariadenia. V práci budú podrobnejšie popísané a vysvetlené problémy, s ktorými som sa stretol počas vývoja aplikácie.

Na začiatku sa budem zaoberať témou čo je to rozšírená realita v samotnom základe. Kedy, ako a prečo vznikla. Kde všade je možné ju využiť, že nepatrí len do pracovného či vývojového prostredia, ale je možné ju využiť aj v zábavnom odvetví. Ďalej ako rôznorodo môže byť nápomocná a ako v niektorých prípadoch ušetrí veľa času, a úsilia. Aké rôzne zariadenia boli vyvinuté, čo znamená skratka HMD a prečo je potrebné navrhnuť správne a precízne používateľské rozhranie. Ďalej sa budem zaoberať systémom ARTable, s ktorým bude aplikácia spolupracovať. No a nakoniec vysvetlím princíp samotného sledovania obrazu (ang. Camera tracking) pomocou kamery, ktorá je prichytená k zariadeniu.

V ďalších častiach práce sa zameriam na samotné vývojové prostredie Unity. Na balík ARToolKit, ktorý som použil na vytvorenie práce a prečo práve tento. Ako funguje program a aké sú jeho implementačné časti. Nakoniec spomenie ako som testoval samotný program a aký je výsledok práce.

Kapitola 2

Teoretická časť

V tejto časti sa budem zaoberať témou, čo je to rozšírená realita, jej históriou a využiteľnosťou v rôznych oblastiach. Ďalej vývojovým prostredím, v ktorom bude práca vytvorená a aké rôzne zariadenia existujú na zobrazenie rozšírenej reality. Nakoniec popíšem aj niečo o ARTable s ktorým budem pracovať.

2.1 Rozšírená realita a jej história

Rozšírená realita je integrácia digitálnych informácií s reálnym prostredím používateľa v reálnom čase. Na rozdiel od virtuálnej reality, ktorá vytvára úplne umelé prostredie, rozšírená realita využíva existujúce prostredie (reálny svet) a obohacuje ho o nové grafické virtuálne počítačom vygenerované informácie. Jednou z dôležitých súčastí je vytvoriť aplikáciu s čo najmenšou možnou odozvou, aby sa všetko odohrávalo v reálnom čase.

AR aplikácie sú naprogramované v špeciálnych 2D alebo 3D vývojových programoch, ktoré umožňujú vývojárom skombinovať animačné či textové prvky digitálneho sveta spolu s reálnym svetom.

AR aplikácie pre smartfóny typicky využívajú GPS (global positioning system) na určenie polohy zariadenia a kompas na určenie orientácie zariadenia. Sofistikované AR programy využívajú rozpoznávanie objektov a rozpoznávanie gest.

Výskumník spoločnosti Boeing, Thomas Caudell ako jeden z prvých zaviedol termín „augmented reality“ v roku 1990. Jednou z prvých komerčných aplikácií AR technológie bola žltá „first down“ čiara, ktorá sa začala objavovať počas futbalových zápasoch v televízií niekedy okolo roku 1998. Google glass spolu s HUD (heads-up display) v automobiloch na predných sklách sú asi najviac známe AR produkty dostupné pre verejnosť. Okrem iného sa AR technológia používa v rôznych odvetviach vrátane zdravotnej starostlivosti, verejnej bezpečnosti, cestovného ruchu a marketingu.

Začiatok rozšírenej reality sa objavil na prelome 60. a 70. rokoch 20. storočia. V tomto období sa začali vyvíjať technológie, ktoré sa dnes používajú pre virtuálnu a rozšírenú realitu. Ide o rôzne snímače, gyroskopy či šošovky. Názov rozšírená realita sa začal objavovať na začiatku 90. rokov, ale až v dnešnej dobe sa dostávame do bodu, kedy je možné rozšírenú realitu sprístupniť aj verejnosti, pretože boli vyvinuté cenovo dostupné technológie. Rozšírená realita spolu s virtuálnou realitou je na úplnom začiatku vývoja a ešte stále nie je využitý jej potenciál [9].

Rozšířená realita bola prvýkrát použitá pre vojenské, priemyselné a lekárske účely. Od roku 2012 sa použitie rozšírilo aj do oblasti zábavy, vzdelávania a ďalších odvetví. Od roku 2016 sa začali vyrábať už aj mobilné zariadenia podporujúce rozšírenú realitu.

2.2 Využitie

V dnešnej dobe už má rozšírená realita využitie v rôznych oblastiach či už vo forme zábavy alebo vývoja. Tu je predstavenie niektorých z nich [1]

Robotika

Rozšírená realita spolu s robotmi vytvára veľa nových možností. Jednou z nich je napr. logistika. Ten najlepší príklad je, keď skladník potrebuje naložiť alebo vyložiť tovar. Skladník s vozíkom ktorý riadi, vidí v okuliariach presnú cestu k tovaru resp. ho okuliare navigujú k presnej polohe tovaru. Skladník taktiež vidí ďalšie úlohy, ktoré má vykonať. Úlohy sú pritom zadávané buď podľa toho, ktorý zo skladníkov je najbližšie, poprípade ktorý je najmenej vyťažený.

Ďalšou z možností je pomocou rozšírenej reality naučiť robota vykonávať nejaké úkony. Robot má k sebe pripevnenú kameru, ktorá mu umožňuje vidieť. Ďalej obsahuje potrebný software. Následne je možné pomocou rozšírenej reality, ktorú vidí človek a robot zároveň, naučiť robota nejaké úkony. Od tých najjednoduchších napr. ako preložiť pomocou jeho robotickej paže objekt z bodu A do bodu B, až po tie zložité, kedy je robot schopný montáže celého zariadenia (napr. klávesnica, časti auta, atď.). Zaujímavým príkladom je robot, ktorý je zabalený do zeleného plátna, na ktorého sa skrz AR zariadenie pozerá človek a vidí, akoby pred sebou mal ďalšiu osobu, ktorej sa môže dotknúť a interagovať s ňou. Robot sa pritom snaží snímať pohyby a gestá osoby a reagovať na nich, akoby reagoval skutočný človek.

Robotika zasahuje aj do zábavného priemyslu. Jedným z príkladov je, že robot v tvare pavúka je skutočný a na displeji mobilu sa zobrazujú prvky rozšírenej reality a tie niekoľko násobne obohacujú zážitok z hry. Robot sa ovláda pomocou aplikácie na mobilnom telefóne a tá súčasne zobrazuje rôzne úlohy, eventy a nepriateľov v podobe AR.

Na univerzite sa nachádza robot s názvom PR2¹. Jednou z aplikácií pre robota je, že robot sa dokáže naučiť jednoduchým úkonom cez AR, ktorá je premietaná na stôl nachádzajúcim sa pred ním. Premietanie AR zabezpečuje projektor a snímanie premietaného obrazu zabezpečuje kinect. Užívateľ zadá robotovi inštrukcie, ktoré následne robot vykoná. Jedným z úkonov môže byť napr.: Zober tento objekt, otoč ho o 180 stupňov a polož ho naspäť na stôl. Robot dokáže objekty premiestňovať, otáčať a rôzne s nimi manipulovať.

Videohry

Rozšírená realita v oblasti video hier prináša hráčom nový zážitok kedy je reálne prostredie obohatené o digitálne prvky. Hráči tak môžu zažiť oveľa intenzívnejší pocit z hrania ako len v bežnom virtuálnom svete. Jednou z najpopulárnejších hier bola v rokoch 2016 hra Pokémon GO (obr. 2.1)², ktorá využívala rozšírenú realitu.

¹<http://www.willowgarage.com/pages/pr2/overview>

²<http://www.newyorker.com/tech/elements/pokemon-go-will-make-you-crave-augmented-reality>



Obr. 2.1: Použitie AR cez mobilnú aplikáciu Pokémon GO



Obr. 2.2: Namodelovaný pôdorys domu

Architektúra

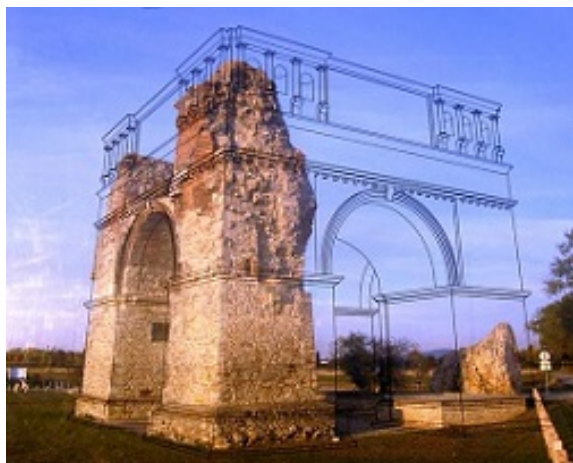
Rozšírená realita v architektúre pomáha pri vizualizácii stavebných projektov. Počítačom generovaný obraz je možné zobrazíť na reálne miesto, kde bude budova ešte len postavená. Rozšírená realita môže byť tiež použitá v pracovnom priestore architekta, kde jeho 2D výkresy sú namodelované do 3D podoby (obr. 2.2)³. Pri prezentácii budovy je možné s použitím rozšírenej reality navrhnuť exteriér či interiér a jeho rozloženie. S neustálym zlepšovaním presnosti GPS (Global position system) je možné vizualizovať rôzne objekty nachádzajúce sa v zemi alebo stenách samotných budov ako elektrické káble alebo vodovodné potrubie.

Archeológia a turizmus

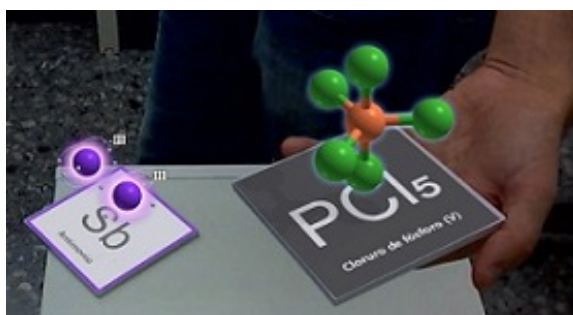
Rozšírená realita bola použitá aj na pomoc pri archeologických objavoch. Pomocou aplikácie je možné namodelovať a zobrazíť ako daná budova, krajina alebo objekt vyzerali za pomoci pozostalostí z daného obdobia (obr. 2.3)⁴. Turisti si tak môžu pomocou zariadenia zobrazíť ako daná budova vyzerala pred niekoľkými storočiami prípadne kto v tejto budove žil a ďalšie informácie o nej. To dokáže priniesť omnoho väčší zážitok, napríklad pri exkurzii.

³<http://www.augment.com/augmented-reality-architecture/>

⁴<https://cz.pinterest.com/pin/367184175841159168/>



Obr. 2.3: Doplnenie zrúcanej stavby do pôvodne vyzerajúcej



Obr. 2.4: 3D modely chemických zlúčenín

Vzdelanie

Vo vzdelávacom prostredí sa do učebníc pridávajú rôzne markery, ktoré sa používajú na doplňujúce informácie k štúdiu. Pomocou aplikácie je nasnímaný nejaký objekt či osoba z histórie a na displeji sa zobrazujú informácie. V chémii je možné nasnímať nejaký vzorec a ten je následne vykreslený na displej v jeho 3D podobe (obr. 2.4)⁵ spolu s informáciami. V oblasti anatómie učiteľ dokáže zobrazíť skryté anatomické štruktúry ako orgány alebo kosti každej osoby v učebni. Astronómia zase umožňuje zobrazovať rôzne súhvezdia a mená hviezd.

Lekárstvo

V oblasti lekárstva pomáhajú rôzne zariadenia lekárom napríklad pri operáciách alebo pri informáciách o pacientoch. Od roku 2005 sa začalo častejšie používať zariadenie, ktoré zobrazuje podkožné žily a premieta ich na pokožku snímaného pacienta (obr. 2.5)⁶. Toto zariadenie sa nazýva: Vein Finder.

⁵<http://www.digitalvmagazine.com/en/2014/01/21/la-plataforma-zientia-facilita-el-aprendizaje-en-las-aulas-utilizando-la-realidad-aumentada/>

⁶<https://www.youtube.com/watch?v=-23o7oYQ2DU>



Obr. 2.5: Vein Finder



Obr. 2.6: Zobrazenie bodov záujmu na bojovom poli

Armáda

Armáda je jedna z prvých oblastí, ktoré začali rozšírenú realitu využívať. Aplikácia umožňuje premietat informácie, ktoré sú mu nápomocné, na prilbu vojaka, HMD či iné AR zariadenie ktoré vojak používa. Dokážu zobraziť kde sa nachádzajú spojenci, kto je nepriateľ, kde sa nachádza pre neho dôležitý bod záujmu (obr. 2.6)⁷, koľko má nábojov v zbrani, atď.

Navigácia

Rozšírená realita môže zvýšiť efektivitu navigačných prístrojov. Informácie môžu byť zobrazované priamo na čelné sklo v automobile a tak indikovať stav vozovky, počasie, cieľový bod a vzdialenosť od neho, aktuálnu rýchlosť vozidla, zaradenie sa do správneho jazdného pruhu, upozornenie na prípadne nebezpečenstvo atď. Podobne môžu byť zobrazované informácie aj pre námorné lode alebo lietadlá či helikoptéry. Pre bežné používanie je možné použiť navigáciu v okuliaroch a do nich je priamo zobrazovaná cesta aktuálnej polohy k cieľovému bodu (obr. 2.7)⁸.

Preklad

Rozšírená realita môže tiež slúžiť ako prekladač kedy sa nasnímaný text zobrazuje na okuliaroch v jazyku ktorému ich nositeľ rozumie. Tak isto to pracuje v prípade hovoreného slova, kedy okuliare či iné zariadenie zobrazujú preložené slovo v podobe textu na displej/sklíčka.

⁷<http://www.dailymail.co.uk/sciencetech/article-2640869/Google-glass-war-US-military-reveals-augmented-reality-soldiers.html>

⁸<http://bit.ly/2qnmyO7>



Obr. 2.7: Navigácia zobrazovaná v okuliaroch

2.3 Zariadenia

Existuje množstvo zariadení, ktoré dokážu reprezentovať rozšírenú realitu. Vo väčšine prípadov sú to okuliare avšak ich technický názov je „head-mounted display“ (HMD) alebo tiež „helmet-mounted display“ [2]. HMD je zariadenie, ktoré je umiestnené pred respektíve na okuliaroch a okrem iného má v sebe umiestnenú kameru, ktorá sníma okolitý reálny svet.

Rozšírenú realitu je možné vytvoriť na takmer akomkoľvek zariadení, ktoré obsahuje kameru napr. telefóny a tablety. V dnešnej dobe už existuje veľké množstvo aplikácií, ktoré rozšírenú realitu na tieto zariadenia podporujú.

Existujú dva zobrazovacie metódy, ktoré dokážu rozšíriť reálny svet. Prvou z metód je 3D zobrazovanie, tá dokáže pracovať s 3D modelmi a rozširovať realitu o 3D modely na rozdiel od druhej metódy, ktorá dokáže reprezentovať len 2D modely. V nasledujúcej časti popíšem HMD a predstavím niektoré zo zariadení pre zobrazovanie v 3D a 2D.

Head-mounted display

Head-mounted display (skr. HMD) je zariadenie, ktoré je umiestnené na hlave. Vpredu sa nachádza optický displej, ktorý je pre jedno oko (monocular HMD) alebo sú dva, pre každé oko zvlášť (binocular HMD). Displej je zväčša typu LCD (liquid crystal displays) alebo OLED (organic light-emitting diodes). HMD sa líšia podľa toho, či dokážu zobrazovať len virtuálnu realitu (počítačom generované snímky) alebo snímky reálneho sveta spolu s tými virtuálnymi, čiže rozšírenú realitu. HMD tvorený zo sklíčok alebo iného priehľadného materiálu (aby bolo možné pozorovať reálny svet), na ktoré sú zobrazované virtuálne prvky, spadajú do kategórie pre AR. HMD pre virtuálnu realitu majú neprehliadnu prednú časť, na ktorej je pripevnený displej a zväčšovacie šošovky. Na obr. 2.9⁹ je možné vidieť okuliare pre AR a na obr. 2.8¹⁰ pre VR (virtuálnu realitu).

3D zobrazovacie zariadenia

Microsoft HoloLens

Toto zariadenie je vyvíjané spoločnosťou Microsoft. Názov HoloLens je odvodený od slov Holograph a Lenses. Už z názvu je možné odvodiť, že okuliare prezentujú a zobrazujú hologram. Ten je interpretovaný do reálneho sveta a viditeľný pomocou týchto okuliarov. Microsoft HoloLens vytvára realitu, ktorú Microsoft pomenoval „Mixed reality“ (Zmiešaná realita), tá zobrazuje virtuálnu realitu do reálneho sveta. HoloLens je možné ovládať gestami a hlasom. Pohľad skrz túto rozšírenú realitu umožňuje zobraziť hologram miesta či

⁹<https://www.vuzix.com/images/news/products/STAR1200XL1200XLD.jpg>

¹⁰<https://arstechnica.com/gaming/2016/10/best-vr-headset-2016-psvr-rift-vive/>



Obr. 2.8: Oculus Rift

mesta (ako napr. Tibet) a týmto hologramom rôzne manipulovať (približovať, oddaľovať a pohybovať 3D hologramom všetkými smermi).

META

META okuliare sú ďalším z produktov, ktoré dokážu zobrazovať 3D modely do reálneho sveta. Okuliare sa ovládajú gestami rúk a majú veľké pole snímania – až 90 stupňov. Okuliare sú navrhnuté prevažne na využívanie v interiéri.

Vuzix STAR 1200 XL

Na záver predstavím zariadenie Vuzix, ktoré je pre mňa najdôležitejšie, pretože na tomto zariadení sa bude testovať tento projekt. Zariadenie Vuzix je vyvíjané spoločnosťou Vuzix Corporation. Vuzix STAR 1200 XL je priehľadné, rozšírenú realitu zobrazujúce zariadenie, ktoré ma binokulárne video zobrazenie [5]. Kamera dokáže snímať 35 stupňový uhol, ktorý je dostačujúci na využitie v priemyselnom a zábavnom odvetví. Okuliare podporujú nielen 3D zobrazenie ale ak to nie je potrebné, dokážu zobrazovať aj 2D modely. Vuzix STAR zaistuje možnosť pripojenia pomocou VGA. Ku komponentom patrí „3DOF motion tracking“ čo je senzor na sledovanie pohybu, zabudovaná kamera na sledovanie a rozpoznávanie skutočného sveta a displej, ktorý umožňuje 3D počítačom generovaný obsah premietat na sklíčka a prekryvať tak reálny svet s tým virtuálnym.



Obr. 2.9: Vuzix STAR 1200 XL

2D zobrazovacie zariadenia

Google Glasses

Toto zariadenie zobrazuje informácie len v podobe 2D. HMD sa nachádza na pravom oku a rozširuje realitu len pre toto oko. Google glasses zobrazujú základne informácie podobne ako

smartphone napr. ako dátum a čas, aktuálne počasie, prichádzajúci hovor prípadne body záujmu. Ovládanie je pomocou hlasu prípadne pohybom po dotykovej ploche umiestnenej na okuliároch. Zariadenie bolo vyvíjane spoločnosťou Google a dnes sa už nepredáva.

Vuzix M100

Tento typ okuliarov podobne ako Google glasses zobrazuje informácie priamo pred vás pomocou jedného displeja umiestneného na pravej strane. Vuzix M100 tiež obsahuje kameru na snímánie okolitého reálneho sveta. Tieto okuliare sú založené na OS Android preto sú kompatibilné so stovkami Android aplikácií. Zameranie týchto okuliarov je prevažne v oblasti podnikania, marketingu a lekárstva.

GlassUP

Základnou myšlienkou pri výrobe týchto okuliarov bolo preniesť základne informácie zo smartfónu priamo pred oči. Displej zobrazuje informácie v podobe správ a notifikácií, ktoré sú jednofarebné aby sa šetrila výdrž batérie. Okuliare sa nedajú kontrolovať pomocou hlasu či postranného touchpadu ako pri ostatných a taktiež neobsahujú žiadnu kameru.

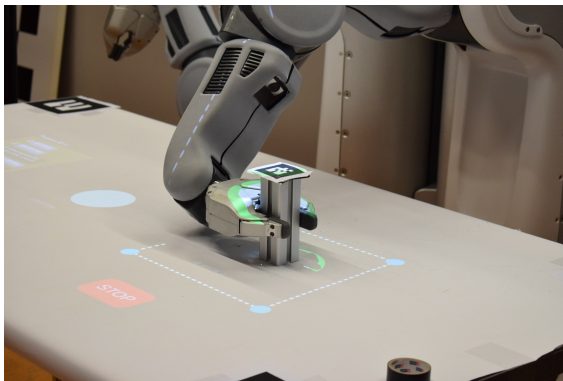
2.4 Používateľské rozhranie

Dôležitou súčasťou pri návrhu používateľského rozhrania (ang. User Interface, skr. UI) je aby nepôsobil rušivo či už na okuliároch, mobiloch alebo iných zariadeniach podporujúcich rozšírenú realitu. Je potrebné ho navrhnuť tak, aby informácie ktoré podáva do zariadenia boli jasné a presné. Najlepšou možnosťou pre zistenie je testovanie rôznymi ľuďmi ktorý, subjektívne zhodnotia ako na nich UI pôsobí.

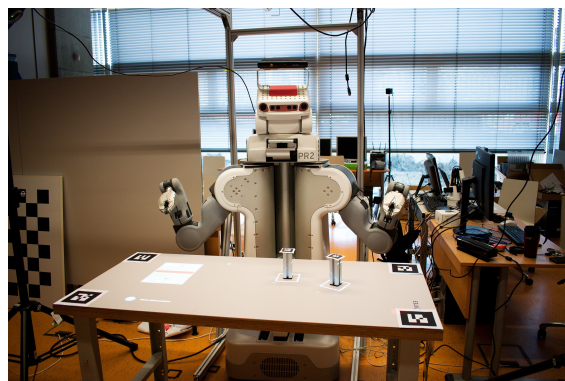
Používateľské rozhranie v rozšírenej realite určite patrí do kategórie nezvyčajných. Je to z toho dôvodu, že nezáleží na samotnom výzore aplikácie kvôli čomu spadá do tejto kategórie, ale na tom, že aplikácia sa mení vzhľadom na okolitý reálny svet a čiastočne sa dokáže prispôbovať.

Rozšírená realita a s ňou používateľské rozhranie poskytuje perfektnú príležitosť pre obohatenie a zlepšenie používateľskej skúsenosti. Ak mám vysvetliť prečo, predstavte si situáciu, že mechanik lietadla, ktorý kontroluje lietadlo či už pred samotným štartom alebo robí bežnú prehliadku celého lietadla. Na miesto toho, aby zakaždým robil fotografie, či video záznamy každej súčiastky. Robil si poznámky na papier alebo to priamo zapisoval do počítača. Zisťoval informácie o každej súčiastke, ako dlho je v prevádzke, prípadne či už bola menená a z akého dôvodu. Namiesto týchto úkonov si nasadí head-mounted-display alebo použije iné zariadenie podporujúce rozšírenú realitu a všetky tieto informácie, ktoré potrebuje sa priamo zobrazujú pred jeho očami a je schopný zvýšiť nielen produktivitu ale hlavne predísť že niečo prehliadne a predísť nejakej nehode [7].

Rozšírená realita je zásadným predchodcom virtuálnej reality. Na rozdiel od virtuálnej reality, ktorá je tvorená len z virtuálnych prvkov, modelov a objektov, tak rozšírená realita spája reálny svet s virtuálnymi prvkami a vytvára tak novú príležitosť nie len v obohatení pracovnej činnosti ale aj v odvetví zábavného priemyslu.



Obr. 2.10: Robotická paža robota PR2



Obr. 2.11: ARTable

2.5 ARTable

ARTable je systém niekoľkých zariadení medzi ktoré patrí: dotykový stôl, robot, kamery, kinect, projekторы. Na stole sa nachádza niekoľko markerov na tento stôl je premietaný obraz v podobe projektorov, a ten rozširuje resp. obohacuje tento stôl o informácie, s ktorými je možné následne pracovať. Tieto informácie dokáže pomocou kamery nasnímať kinect, a následne ich zašle na vyhodnotenie. Robot s menom PR2 podľa vyhodnotených správ vykonáva prislúchajúce úkony.

Tieto úkony je možné uplatniť v menších a stredne veľkých priemyselných firmách, kde môžu vykonávať opakujúcu sa činnosť bez prítomnosti človeka a ten sa môže zamerať na iné dôležitejšie činnosti čím sa zvýši produktivita práce.

Robota je možné jednoducho naprogramovať/preprogramovať pomocou rozšírenej reality zobrazujúcej sa na stole. Kde je schopný pomocou kamery nasnímať čo má vykonávať. Používateľ v podstate naučí robota, akú úlohu má vykonávať a ako ju má vykonávať.

„Používateľ naučí robota spolupracovať a asistovať pri montáži vozíka na obsluhu lietadiel. Je dôležité robotovi ukázať, ktoré časti sú potrebné v každom kroku montáže, kde musia byť vyvrtané otvory a aké časti by mali byť spojené dohromady.“ [8]

Robot PR2 (obr. 2.10) je jedna zo súčastí ARTable (obr. 2.11). Obsahuje kamery a dve mechanické paže, ktoré sú schopné zo stola umiestneného pred ním uchopiť rôzne veci a manipulovať nimi (dokáže ich premiestňovať a otáčať ich).

2.6 Camera tracking

Jednou z najdôležitejších častí v rozšírenej realite je lokalizácia kamery, hlavne v prípade head-mounted display (HMD). Na lokalizáciu je potrebné, aby kamera rozpoznala kde sa nachádza vzhľadom k okolitým objektom reálneho sveta. Takéto rozpoznávanie nie je jednoduché, preto sa používajú značky – markery. Markery sú objekty umiestnené v scéne. Majú zvyčajne čierno-bielu farbu. Samozrejme je možné použiť marker akejkoľvek farby, no dôležitý je tvar a nie farba. Čierno biela farba sa používa z dôvodu najvýraznejšieho kontrastu. Markery majú rôzne vzory a sú zväčša charakteristicky podobné QR kódom. Doposiaľ existujú tieto dva základne druhy rozpoznávania: Marker-based a Markerless system.

Marker-based system

Táto technológia používa značky (markers) ako referenčný bod podľa ktorého sa zariadenie dokáže lokalizovať [3]. Zariadenie musí danú značku poznať, aby ho dokázal lokalizovať. Jednou z možností je využiť marker tak, že po lokalizácii sa zobrazí nad touto značkou napr. nejaký 3D objekt. Ďalšou možnosťou je použiť značky na označenie nejakej plochy (napríklad stolu). Označiť ho značkami po rohoch a reprezentovať tak danú označenú plochu pre zariadenie ako vymedzenie určitej oblasti na ktorej sa môže niečo diať.

V systéme založenom na značkách je oveľa jednoduchšia implementácia, pretože je vopred dané čo má značka vykresliť či zobrazíť na zariadení. Na rozdiel od zariadenia, ktorý nepoužíva značky je potrebné implementovať algoritmus, ktorý rozpoznáva okolie. Napríklad, ak je algoritmus schopný identifikovať psov, tak to znamená, že aplikácia AR bude môcť zobrazovať rozšírené informácie kedykoľvek je identifikovaný pes, bez toho aby boli všetky rasy psov na svete implementované kódom pri vývoji aplikácie.

Markerless system

Systém bez použitia symbolov sa väčšinou využíva v smartfónoch a tabletoch. Táto metóda využíva kombináciu akcelerometra, kompasu a lokalizačných dát (napríklad GPS – Global Position System) zo zariadenia na určenie polohy oproti reálnemu svetu [3]. Tieto dáta môžu byť porovnané z databázou pre určenie na čo sa zariadenie práve pozerá a tým povolí zobrazenie rozšírených informácií na obrazovku zariadenia. V dnešnej dobe existujú aplikácie na operačný systém Android, iOS, Windows a ďalšie.

Kapitola 3

Návrh riešenia

Predtým ako začnem s popisom, vysvetlím zásadný pojem objekt a inštancia. Objekt môže akýkoľvek objekt nachádzajúci sa na stole. Inštancia je virtuálny prvok (kruh, polkruh, dodatočné informácie k objektu) vytvorený počítačom, ktorý bude obohacovať reálny svet.

Cieľom práce je vytvoriť rozšírenú realitu pre ARTable, na ktorom sa budú nachádzať rôzne objekty. Okolo týchto objektov sa za pomoci okuliarov pre rozšírenú realitu zobrazia kruhy (viditeľný len pomocou okuliarov), ktorý bude symbolizovať kde sa daný objekt nachádza. Jednou z možností je vykresľovať okolo objektu bounding box, čo je najmenší možný štvorec/obdĺžnik v 2D resp. kocka/kváder v 3D, ktorý je možné nad daným objektom vytvoriť. Ďalej zobrazenie rôznych informácií o objekte, ako je jeho názov a informácie o tom, kde sa objekt nachádza. Aby bolo možné určiť polohu objektu na stole je potrebné vytvoriť súradnicový systém. Ten je vytvorený pomocou markerov nachádzajúcich sa na stole, čo umožní okuliarom orientáciu na tomto stole a lokalizáciu okuliarov voči stolu v priestore.

Prvým krokom je získať informácie od ARTable systému. Tieto informácie/dáta preniesť do reálneho sveta resp. AR podporujúceho zariadenia. Dáta sú zobrazované za pomoci okuliarov na jednej zo súčasti ARTable – na stole. Ako programovací nástroj som použil Visual Studio 2015, a ako vývojový program herný engine Unity. Ako doplnok k Unity som použil ARToolKit, ktorý bol vyvinutý pre prácu s rozšírenou realitou. Dokáže rozpoznať dva základné markery (Hiro a Kanji), ktoré som použiť pri počiatočnom návrhu implementácie, keď som sa pokúšal vytvoriť základné 3D objekty nad týmto markerom. Následne som sa pokúšal pracovať s oboma markermi súčasne. Vytvoril som medzi nimi čiaru, ktorá sa zobrazovala, keď boli obidva markery vo viditeľnom poli okuliarov Vuzix resp. kamery, ktorá je na nich prichytená. Na nasledujúcich obrázkoch je možné vidieť dva základné markery.



Obr. 3.1: Hiro



Obr. 3.2: Kanji

Ďalšou úlohou bolo vytvorenie komunikačného kanálu s Robotom, ktorý mi bude poskytovať informácie o objektoch nachádzajúcich sa na stole. K tomu som použil jednu z knižníc a to konkrétne ROSBridgeLib. Informácie od robota boli poskytované pomocou JSON formátu. Na jeho analýzu som použil knižnicu SimpleJSON. Pomocou výstupného formátu z tejto knižnice som sa mohol dostať ku každému potrebnému parametru. Server mi podával informácie o tom, kde sa nachádza objekt na stole t.j. jeho pozíciu. Ďalej rozmery objektu a v neposlednom rade aj názov samotného objektu.

Predposledný krokom bolo vytvorenie kruhu okolo daného objektu nachádzajúceho sa na stole. K problému, na ktorý som počas testovania tohto kruhu narazil bolo, že inštancia kruhu pokrývala samotný objekt umiestnený na stole a výsledný efekt bol mätúci pre používateľa okuliarov. Problém vznikol, pretože inštancia kruhu bola jediná, ktorá bola definovaná. O objekte, ktorý bol na stole som v mojom programe nesmel mať žiadnu informáciu, preto som nedokázal vytvoriť „ilúziu“, kedy kruh v určitých častiach nie je viditeľný, t.j. kedy je pokrývaný objektom na stole. Problém som vyriešil tým, že som kruh zmenil na polkruh. Okrem polkruhu som vypisoval aj informácie o objekte a to konkrétne jeho pozíciu a názov.

Posledným krokom bolo vytvorenie viacnásobného markeru alebo viacerých markerov. Na začiatkoch práce som používal len s jeden marker (konkrétne Hiro), ktorý bol umiestnený v ľavom dolnom rohu a definoval okrem iného aj začiatok súradnicového systému $x,y,z = [0,0,0]$. Pri používaní len jedného markeru vznikol problém, keď som sa na neho cez okuliare pozeral z veľmi ostrého uhlu prípadne som ho začal strácať v poli viditeľnosti kamery. Zapríčinilo to to, že neboli vykresľované žiadne inštancie, pretože program nevedel na aké miesto ich má vykresliť, keďže nemal marker v zornom poli. Prvým nápadom bolo použiť Uniform Marker Fields - UMF marker [6] (obr. 3.3). Silnou stránkou tohoto markeru je to, že pokrýva celú plochu stola. Tým sa teda stráca problém, že by marker nebolo vidno. Ďalšou výhodou je, že marker môže byť pokrývaný objektami umiestnenými na stole. Nastáva teda otázka prečo nepoužiť marker Hiro zväčšený cez celú plochu stola. Tam ale nastáva ďalší problém. Ak by bol teda marker Hiro extrémne zväčšený a pokrýval by celú plochu stola, tak by nebolo možné tento marker identifikovať z dôvodu, že by bol prekrytý objektami na stole. Tento marker totiž nemá rovnaké vlastnosti ako UMF marker. Kamera by následne nemohla marker lokalizovať a nebolo by možné nič do scény vykresľovať. Tento problém ale UMF rieši a môže byť pokrývaný objektami ako som už spomínal.

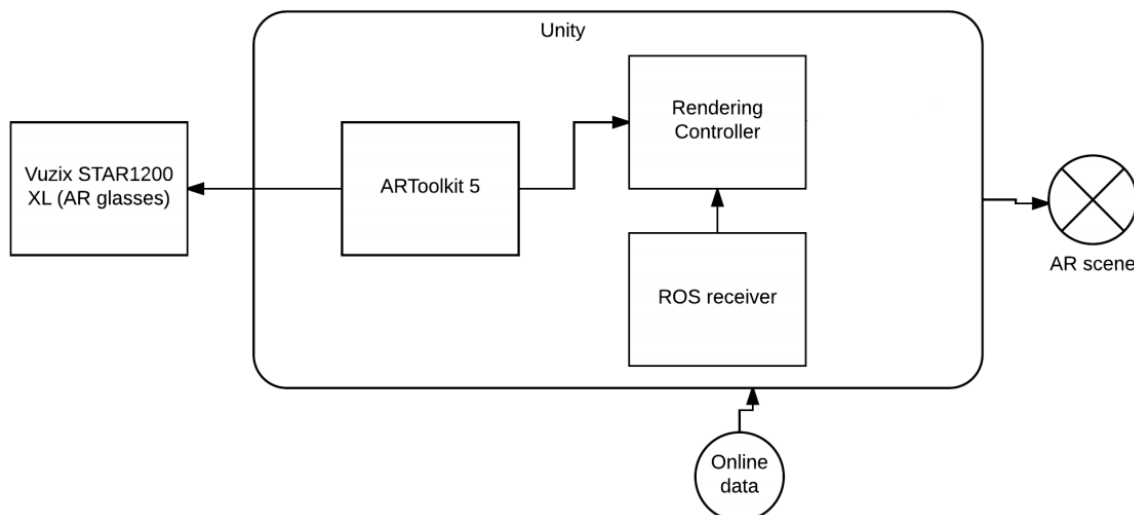


Obr. 3.3: UMF marker



Obr. 3.4: Objekty pokrývajúce UMF marker

Problém nastal, že by bolo príliš náročné implementovať UMF marker vo fáze projektu, v ktorej som sa už nachádzal. Preto som zvolil alternatívne riešenie a to použiť Multimarker od Unity. Multimarker je zhluk viacerých markerov na ploche, kde každý marker má svoju



Obr. 3.5: Základný koncept systému

polohu. Túto polohu je potrebné definovať pomocou špeciálneho súboru. V tomto súbore sa nachádzajú informácie o tom koľko bude registrovaných markerov, aká je ich vzájomná poloha a akú majú veľkosť v milimetroch. Okrem toho je potreba definovať transformačnú maticu pre každý marker. Existujú tri verzie Multimarkera. Prvou z nich je verzia kocky, ktorá obsahuje jeden marker na každej stene kocky. Druhou je verzia 16 markerov a treťou je verzia 64 markerov. V mojom programe som použil verziu s šiestimi markermi.

Na obrázku obr. 3.5 je vidieť základný koncept systému so všetkými jeho časťami. Knižnica ARToolkit bude komunikovať s okuliarmi a pomocou kamery nasníma scénu. ROS receiver prijme dáta, ktoré zaslal robot a zašle ich na vykreslenie. Spolu s dátami od ARToolkit a kamery, ktorá nasníma scénu je možné vykreslenie výslednej scény.

3.1 Popis systémových častí

Celý systém pozostáva z niekoľkých častí: okuliarov, markerov, kontrolóra ktorý vykresľuje informácie získané od robota resp. serveru. V nasledujúcej časti popíšem všetky časti podrobnejšie.

Unity

Vývojovým prostredím pre túto prácu bude Unity vyvíjané spoločnosťou Unity Technologies. Unity je prevažne herný engine (ang. game engine) určený na vývoj video hier na počítač, konzolu či mobil. Unity je možné používať na operačných systémoch Windows, Linux alebo OS X a podporuje vývoj na rôzne platformy (Android, Playstation, Xbox, atď.). Podľa toho na akom OS sa vyvíja, tak sú používané rôzne grafické knižnice. Pre Windows to je Direct 3D, pre Linux a OS X je to OpenGL.

Na začiatok je dobre sa oboznámiť a vysvetliť ako Unity pracuje. Unity využíva skripty. Tie môžu byť písané v jazyku C#, JavaScript alebo Boo. Ja som si zvolil C#. Ďalšou dôležitou informáciou je, že skripty sa prichytávajú k objektom. Keďže Unity je herný engine, väčšinou sú objekty definované ako GameObject, čiže herný objekt. To ale nijak neovplyvňuje inštalácie, z ktorými budem pracovať. V neposlednom rade je potreba si uvedomiť,

že všetky skripty sa spúšťajú súčasne a paralelne. V Unity je zložité vytvorenie „threads“, pretože je to vyhodnocované prekladačom ako chyba.

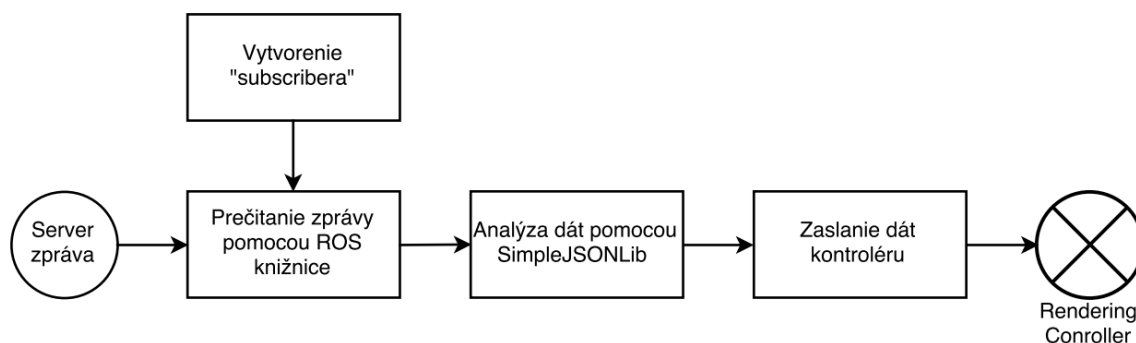
Dôležitou súčasťou Unity je „Assets store“, ten ponúka rôzne balíky, ktoré je možné použiť na vývoj iných aplikácií. Tieto balíky sú vyvíjane buď samotnými vývojármi Unity alebo nezávislými vývojármi, ktorí sa radia medzi komunitu. Balíky sú buď voľne dostupné alebo je potrebné za nich zaplatiť. V každom prípade je potrebné rešpektovať licenčné podmienky použitia daného balíka.

Vuzix STAR 1200 XL

Ďalšou dôležitou súčasťou sú samotné okuliare, ktoré sa skladajú z kamery zachytávajúcej okolitý svet, sklíčok a dvoch zariadení (každé na jedno sklíčko), ktoré premietajú výslednú scénu na sklíčka a poskytujú tak pocit rozšírenej reality. Okuliare môžu pracovať na systéme markerless a marker-based. Pre moje potreby som použil marker-based systém, ktorý je dostačujúci a jednoduchší na implementáciu. Okuliare sa dokážu lokalizovať pomocou markera čo znamená, že podľa toho dokážu určiť svoju polohu (napr. ako sú vzdialené od markera).

ROS Receiver

ROS Receiver je program, ktorý zabezpečuje príjem správ do robota. Pre ich získanie je potrebné vytvoriť tzv. „listenera“, ktorý bude naslúchať danú informáciu. Listener je pridaný ako subscriber, ktorý odoberá správy zo serveru. Získane správy sú vo formáte JSON a je potrebné ich analyzovať do súboru, z ktorým môžem ďalej pracovať. Po analýze sa informácie ďalej predávajú ovládaču (ang. controller), ktorý tieto dáta vyhodnocuje a vytvára výslednú scénu zobrazujúcu sa v okuliároch. Na obr. 3.6 je názorne ukázaný postup.

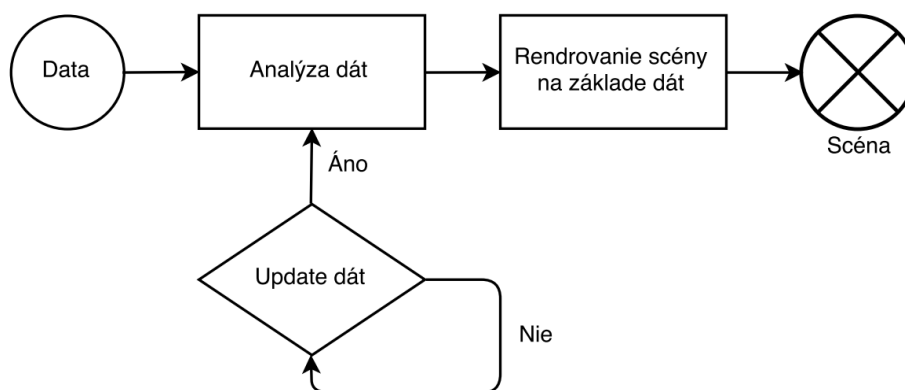


Obr. 3.6: Prijímanie správ od robota

Rendering Controller

Na dosiahnutie cieľa je potrebné vytvorenie programu, ktorý bude schopný komunikovať s robotom a zabezpečiť tak zasielanie potrebných správ. Druhou časťou je vytvorenie programu, ktorý bude schopný vykresliť výslednú podobu projektu. Ja budem pracovať s dvoma inštanciami. Jednou z nich je polkruh a druhou sú informácie o danom objekte umiestneného na stole. Na obrázku 3.7 je vidieť, ako sa postupne prechádza z jedného stavu do druhého a dostáva sa postupne finálneho výsledku. Na začiatku je potreba získať informácie od serveru. To zabezpečuje ROS receiver resp. ROSBridge. Následne je potrebné

dáta rozložiť a analyzovať, to znamená previesť ich z podoby JSON formátu do formátu, z ktorým môžeme ďalej pracovať. Nasleduje analýza získaných informácií, čiže zistenie čo odoslal server a aké informácie sú pre mňa potrebné. Tieto informácie sú predané ďalej a začína samotné vykresľovanie scény na základe potrebných informácií. Poslednou súčasťou je Update. Ten zabezpečuje že pri každom novom obdržanom súbore resp. správe s novými dátami sa proces opakuje a dochádza k opätovnému vykresľovaniu scény. Update je potrebný vzhľadom nato že scéna sa môže meniť. V scéne sa môže počas behu programu objektami pohybovať, môžu byť pridané a odstránené.



Obr. 3.7: Vykreslenie rozšírených informácií

3.2 Výsledná scéna

Výsledná scéna je rozdelená do dvoch vrstiev. Prvá z nich je tá reálna, ktorá zachytáva okolitý svet pomocou kamery na okuliaroch. Druhá vrstva je tá, ktorá obohacuje tu reálnu o rozšírené prvky. V našom prípade o polkruh a informácie o objekte. Výsledný efekt sa zobrazuje v okuliaroch a ich nositeľ vidí obe vrstvy.

Nositeľ AR zariadenia teda vidí vo finále objekty, ktoré sa nachádzajú na stole. Ďalej polkruh, ktorý symbolizuje umiestnenie objektu – polkruh sa nachádza v tesnej blízkosti hneď pod objektom. A nakoniec informácie o objekte, ktoré sa nachádzajú hneď pod polkruhom resp. objektom.

3.3 Súradnicový systém

ARTable má svoj súradnicový systém, ten je reprezentovaný pomocou markerov. Kinect nasníma markery umiestnené v rohoch stola a podľa nich vytvorí tri základné osi x, y, z. Program pre okuliare sa musí s týmto súradnicovým systémom zhodovať, aby nenastal problém pri zobrazovaní inštancií. Tie by sa v nezhodných súradnicových systémoch mohli nesprávne umiestňovať k objektom resp. by boli umiestnené úplne na inom mieste, ako je objekt.

Pri vytváraní zhodného súradnicového systému ako ma ARTable, je potrebné zadať rovnaké merné jednotky a škálovanie (ang. scale). V našom prípade je potreba nastaviť jednotky na milimetre a škálovanie v pomere k veľkosti markeru a scény.

Kapitola 4

Implementácia

Na začiatku bolo potrebné vybrať ako vhodný programovací jazyk tak aj vývojové prostredie pre vývoj programu. Rozhodoval som sa medzi programovacími jazykmi python, C++ a C#. Nakoniec som vybral jazyk C# bolo to z toho dôvodu že tento jazyk bol podporovaný v Unity. Unity som zvolil z toho dôvodu, že pri hľadaní správneho vývojového prostredia som narazil na ARToolKit, čo je knižnica pre Unity, ktorá obsahuje podporu pre rozšírenú realitu. V nasledujúcej časti sa zameriam na podrobný popis jednotlivých implementačných častí.

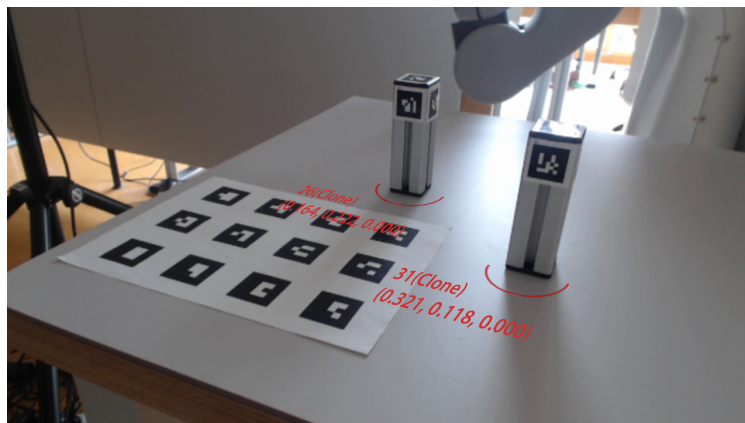
4.1 Implementačné nástroje

Unity

Prvou a základnou stavebnou časťou je vývojové prostredie. Unity je zväčša používaný pri vývoji hier pretože je to tzv. Game Engine. Pre účel vývoja tohto projektu to vôbec nie je problém, pretože ako som už spomínal, tak je pre unity balík s názvom ARToolKit o ktorom sa zmienim podrobnejšie neskôr. V unity sa skripty prichytávajú k objektom (konkrétne GameObject), takže je potrebné mať dopredu vytvorenú inštanciu tohoto objektu, ku ktorému bude tento skript prichytený. Skript definuje inštanciu aké sú jej vlastnosti, ako napr. fyzikálne (pôsobí na neho gravitácia, reaguje na kolíziu s inými predmetmi, atď.), kozmetické (textúra objektu, farba, tvar, tieň, atď.). Objekty sa vkladajú do scény. Všetko čo sa nachádza v scéne je viditeľné pri výslednom spustení projektu. V našom prípade je scéna všetko to, čo je v zornom poli okuliarov resp. kamery prichytenej na okuliároch. Inštancie, ktoré budem pomocou programu vytvárať, budú vykresľované do tejto scény a budú vytvárať rozšírenú realitu.

V unity okrem iného existujú dve základne metódy. Prvou je metóda **Start()** a druhou **Update()**. Je potrebné vedieť že v unity sa všetko spúšťa so spustením programu. Všetky prichytene skripty sa spúšťajú naraz. Metóda **Start()** sa vykoná ako prvá. Metóda **Update()** sa volá automaticky každý snímok (frame). Toto môžem využiť v mojom projekte, pretože potrebujem kontrolovať akékoľvek zmeny týkajúce sa objektov umiestnených na stole.

V mojom projekte potrebujem dve inštancie typu GameObject. Jedena z nich bude reprezentovať polkruh a tá druhá dodatočné informácie k objektu. Tieto informácie sa nachádzajú pod objektom. Keďže sa na stole môže nachádzať viac objektov je potrebné týchto polkruhov a informácií pridať viac-násobne. Na tento prípad sa najlepšie hodia objekty nazývané Prefabs. Každý prefab je vlastne jedna inštancia, ktorá má nejaké základné



Obr. 4.1: Názorná ukážka programu s dvomi objektami a k nim prisluchajúcimi inštanciami

vlastnosti, ako napr. veľkosť, tvar, farbu a i. Tento prefab, môže byť počas behu programu kedykoľvek vložený do scény, to znamená, že ak je na začiatku programu na stole len jeden objekt, vytvorí sa jeden prefab s polkruhom a jeden s informáciami o modeli. Ak sa počas behu programu pridá ďalší objekt, sú vytvorené ďalšie dva prefabs. Opäť polkruh a informácie. Na obr. 4.1 je možné vidieť scénu s dvoma objektami a k nim vygenerovanými inštanciami.

V projekte sú dve základné metódy, ktoré slúžia k vytvoreniu scény. `CreateScene()` a `UpdateScene()`. Prvá z nich slúži k prvotnému vytvoreniu scény. Sú z nej volané ďalšie dve metódy a to `CreateObjects()` a `Informations()`. Tieto dve metódy vygenerujú potrebný počet polkruhov a informácií k modelom. Metóda `UpdateScene()` slúži na to, ak počas behu programu dôjde k nejakej zmene v scéne, ako napr. pohybu objektu po stole. Metóda následne zmení polohu polkruhu a informácií vzhľadom na polohu modelu nachádzajúceho sa na stole.

Ďalšou dôležitou časťou je, že modely môžu byť počas behu programu pridávané a odobierané. Preto je potrebné k nim následne vytvoriť či odobrať polkruh a informácie. Vytvorenie zabezpečuje metóda ktorú som už spomenul: `CreateScene()`. V nej dôjde k porovnaniu dvoch premenných typu – List kde jeden obsahuje staré informácie o objektoch nachádzajúcich sa na stole a druhý nové. V prípade zmien sa prevedú potrebné operácie na zachovanie presnosti vykresľovania scény. Ak dôjde k odobratiu predmetu zo stola je volaná metóda `Destroy()`, ktorá zabezpečuje odstránenie prefabs ktoré sú vykresľované do scény navyše.

ARToolKit

ARToolKit je rozsiahly balík, ktorý obsahuje množstvo skriptov. Ja budem potrebovať niektoré z nich a to konkrétne: `ARController`, `ARMarker`, `AROrigin`, `ARTrackedObject` a `ARCamera`. Predtým ako popíšem samotné skripty je potrebné ukázať hierarchiu systému v unity a načítanie samotného balíku. Na obr. 4.2 je názorná ukážka hierarchie.

Každý nadradený prvok v hierarchii je rodič (parent) a podradený dieťa (Child). Každý prvok môže mať svoje samostatné vlastnosti, no niektoré sa viažu z rodiča na dieťa, a to napr. tak, že ak v scéne nieje viditeľný rodič nebude viditeľné ani dieťa, a to ani prípade, že sa mu nastaví pomocou skriptu viditeľnosť na hodnotu = `true`.

Balíky sa v unity importujú jednoducho a ľahko. Stačí vybrať možnosť „Import New Assets“, a vybrať ktoré časti sa majú importovať, prípadne importovať celý balík.



Obr. 4.2: Hierarchia unity

Teraz sa zameriam na popis skriptov, ktoré som použil v projekte [4]. ARController a ARMarker sú prvé dve základne skripty, ktoré je potrebné prichytiť na najvyššieho parent-a v hierarchii. Pomocou skriptu ARController je možné po spustení programu vidieť v unity video z kamery naživo. ARMarker slúži na definíciu markeru. Ak chcem rozpoznávať viacero markerov potrebujem viacero skriptov – pre každý marker jeden skript. Existuje možnosť multimarker a v tomto prípade prichytávam len jeden ARMarker avšak je potrebné vytvoriť špecifický súbor, ktorý definuje tento marker ako som už spomínal.

Ďalší v poradí je AROrigin ten sa prichytáva na GameObject, ktorého názov je zvyčajne „Scene root“. Prebiehajú v ňom základné transformácie a výpočty. ARTrackedObject je samotný marker v scéne, ktorý bude vyhľadávaný kamerou po spustení programu a bude slúžiť ako záchytný bod napr. pre lokalizáciu kamery. Nakoniec ARCamera skript je prichytený na GameObject Camera v hierarchii. Je to posledná vec ktorú je potrebné urobiť pred tým, ako bude pomocou kamery zachytávaný, viditeľný a rozpoznávaný akýkoľvek kontext.

Dôležitou súčasťou okrem prichytávania je správne nastavovanie viditeľnosti. V unity sa inštancie objektov umiestňujú vo vrstvách (layers). Tie základné sú ARForeground, ARBackground, ARBackground2 a Default. V prípade nesprávneho nastavenia môže nastať problém, s ktorým som sa taktiež v projekte stretol a to, že objekty nebudú zobrazované v scéne aj keď sú viditeľné.

ROSBridge

ROSBridge je ďalšou súčasťou projektu. Jeho úlohou je získanie správ, ktoré odosiela robot. Ako som už spomínal, je potrebné vytvoriť listenera resp. subscribera, ktorý tieto správy bude odpočúvať a posilať ich pre ďalšie spracovanie a využitie. V tomto prípade je potrebné vytvoriť dvoch takýchto subscriberov. Jeden z nich získava informácie o pohybujúcich sa objektoch na stole a ten druhý o nehybných.

Subscriber sa pridáva pomocou metódy `AddSubscriber(typeof())`. Ten potom obsahuje informáciu o tom či odoberá informácie od pohybujúcich sa objektoch alebo o tých nehybných. Nastavenie odoberania zabezpečuje metóda `GetMessageTopic()` ktorá obsahuje `return /objects_string` pri nehybných objektoch a `/objects_string_moving` pri pohybujúcich sa objektoch.

SimpleJSON

SimpleJSON je skript, ktorý zabezpečuje spracovanie správ poskytnutých od robota. Keďže robot zasiela správy vo formáte JSON je potrebné ich analyzovať, aby sa nimi dalo lepšie pracovať. Tuto činnosť vykoná spomínaný SimpleJSON, ktorý umožní prístup k jednotlivým častiam správ už len pomocou vyhľadávajúcich kľúčov.

4.2 Testovanie

Svoju prácu som začal testovať najprv na obyčajnom stole so základný markerom – Hiro. Skúšal som rôzne kombinácie farieb a veľkosti kruhov resp. polkruhov. Nakoniec som sa rozhodol pre červenú farbu, pretože je dostatočne výrazná. Taktiež som použil rovnakú farbu na informácie. Zo začiatku som testoval len na lokálnom súbore. Ten bol typu JSON, aby som mohol otestovať ako mi bude SimpleJSON analyzátor vyhovovať. Následne som vytvoril komunikáciu s robotom a začal získavať informácie o objektoch umiestnených na stole, ktoré odosielať server. Vytvoril som subscribera len pre nehybné modely a potom aj pre pohybujúce sa modely. Nakoniec som testoval program už na systéme ARTable. Pri testovaní na tomto stole nastali problémy so správnou lokalizáciou objektu, kedy sa na stôl pozeralo pomocou kamery na okuliároch v príliš ostrom uhle. Uhol z pravej strany zobrazoval lokalizáciu správne zatiaľ čo ten z ľavej strany nedokázal správne vykresliť inštancie pri jednom z objektov. Na nasledujúcich obrázkoch je možné vidieť dva uhly pohľadu pričom objekty sú stále na tom istom mieste.



Obr. 4.3: Nekorektné zobrazovanie inštancií



Obr. 4.4: Korektné zobrazovanie inštancií

Problém som skúmal dlhšiu dobu. Ako prvé som kontroloval môj kód či nieje chyba nikde vo vykresľovaní prípadne škálovaní inštancií. Následne som urobil kalibráciu okuliarov pomocou programu od Unity Technologies a špeciálneho šachovnicového markeru. Problém ale stále neprestával. Nakoniec som pomocou programu pre Vuzix okuliare zistil že problém je buď v gyroskopoch alebo akcelerometroch, pretože softvér nezobrazoval správne hodnoty pri ich natáčaní. Nemal som k dispozícii ďalšie okuliare, ktoré by som mohol použiť a tým zistiť či program funguje správne a problém je v samotných okuliároch.

Ďalší problém s ktorým som sa stretol bolo, že som nemohol výsledný program otestovať v samotných okuliároch. Problém bol ten že okuliare po pripojení k počítaču sa správali ako ďalší pripojený monitor. Zobrazovali teda plochu monitora. Po spustení programu sa nič nezmenilo a jedinou možnosťou ako testovať program, bolo použiť kameru od okuliarov, a obraz ktorý premietal Unity aj s inštanciami.

Kapitola 5

Záver

Na začiatku práce bolo potrebné zoznámiť sa z rozšírenou realitou. Zistiť ako pracuje a vybrať vhodný programovací nástroj na vývoj aplikácie. Zistil som, že Unity je nástroj na vývoj hier. No dokáže oveľa viac a existuje pre neho veľa užitočných balíkov, ktoré podporujú vývoj rôznorodých aplikácií.

Pomocou Unity sa podarilo vytvoriť aplikáciu, ktorá dokáže okolo objektov vykresľovať rôzne informácie. Zobrazí meno objektu, jeho polohu a označí polkruhom jeho umiestnenie. Dôležitou časťou vývoja bolo vytvorenie zhodných súradnicových systémov, aby nedošlo k nesprávnemu zobrazovaniu inštancií a tým k nepresnému a nevyhovujúcemu výsledku.

Počas programovania som sa stretol s mnohými problémami. Ten najväčší z nich bol, prečo okuliare vykresľovali z jedného uhla pohľadu inštalácie správne a z iného nesprávne. Zabralo veľa času a úsilia dojsť k záveru, že je problém v samotných okuliároch. Ak by som na začiatku práce vedel to čo viem teraz, určite by som skúsil použiť lepší grafický návrh, aby inštalácie vyzerali viac futuristicky. Keďže pre niektorých, je ako rozšírená realita, tak virtuálna realita niečím úžasným, tak by práca v konečnom výsledku mohla mať lepší vzhľad.

V práci by bolo možné pokračovať. Jednou z možností by bolo otestovanie, ako by fungovala aplikácia v prípade že by marker bol vytlačený cez celý stôl. Rozhodnúť či by bolo lepšie použiť verziu s 16-timi markermi alebo tú s 64-mi markermi. Inou možnosťou by bolo vytvorenie ďalšej inštalácie, ktorá by obsahovala informácie o tom, čo práve vykonáva robot PR2. Vypisovať informácie o tom, ako robot manipuluje s objektami. Kam ich presúva, či ako ich má otočiť. Preto je aplikácia navrhnutá tak, aby bolo možné takéto rozšírenia jednoducho pridávať.

Práca bola pre mňa prínosná nielen z pohľadu nových vedomostí, ale aj preto, že som mohol pracovať so systémom ARTable, ktorý ponúka veľa možností a okuliarmi pre rozšírenú realitu, ku ktorými by som inak prístup nemal.

Literatúra

- [1] *Augmented reality*. Wikipedia, [Online; navštívené 12.05.2017].
URL https://en.wikipedia.org/wiki/Augmented_reality
- [2] *Head-mounted display*. Wikipedia, [Online; navštíveno 12.05.2017].
URL https://en.wikipedia.org/wiki/Head-mounted_display
- [3] *What is Augmented Reality*. University of Exeter, Červenec 2011, [Online; navštíveno 12.05.2017].
URL <http://blogs.exeter.ac.uk/augmentedreality/files/2011/07/University-of-Exeter-What-is-AR-Factsheet.pdf>
- [4] *Getting Started with ARToolKit for Unity*. Unity Technologies, Únor 2016, [Online; navštíveno 12.05.2017].
URL https://artoolkit.org/documentation/doku.php?id=6_Unity:unity_getting_started
- [5] Corporation, V.: *Vuzix Announces Advanced STAR 1200 XL See-Through Augmented Reality Glasses*. Vuzix Corporation, Září 2012, [Online; navštíveno 12.05.2017].
URL <http://ir.vuzix.com/press-releases/detail/504/vuzix-announces-advanced-star-1200-xl-see-through-augmented>
- [6] Herout, A.; Szentandrás, I.; Zachariáš, M.; aj.: *Five Shades of Grey for Fast and Reliable Camera Pose Estimation*. FIT VUT v Brně, 2013, [Online; navštíveno 12.05.2017].
URL <http://www.fit.vutbr.cz/~izacharias/papers/2013-CVPR-UMF.pdf>
- [7] Li, A.; Fessenden, T.: *Augmented Reality: What Does It Mean for UX*. Nielsen Norman Group, Září 2019, [Online; navštíveno 12.05.2017].
URL <https://www.nngroup.com/articles/augmented-reality-ux/>
- [8] Materna, Z.; Kapinus, M.; Beran, V.; aj.: *Using Persona, Scenario, and Use Case to Develop a Human-Robot Augmented Reality Collaborative Workspace*. FIT VUT v Brně, 2017, [Online; navštíveno 12.05.2017].
URL <http://www.fit.vutbr.cz/research/pubs/conpa.php.cz.iso-8859-2?file=%2Fpub%2F11301%2Fpersona-scenario-case.pdf&id=11301>
- [9] Zagorová, M.: *Rozšířená realita*. Únor 2012, [Online; navštíveno 12.05.2017].
URL http://wiki.knihovna.cz/index.php?title=Roz%C5%A1%C3%AD%C5%99en%C3%A1_realita

Príloha A

Dodatočné súbory a predmety

CD disk obsahujúci:

- aplikáciu pre okuliare Vuzix STAR 1200 XL,
- bakalársku prácu vo formáte pdf,
- promo video ukážky funkčnosti aplikácie.